Docket No. AUS990800US1

CLAIMS:

What is claimed:

1. A method in a data processing system for managing
5 data attributes, the method comprising the steps of:
registering attributes with a PKCS9 gateway class,
wherein the attributes include user-defined attributes
and PKCS-standard (Public Key Cryptography Standards)
defined attributes; and
10 associatively storing an identifier for each of said
attributes.

2. The method of claim 1 further comprising:
calling a first object-oriented method in the PKCS9
15 gateway class, wherein the first object-oriented method
receives a parameter comprising an object identifier for
an attribute.

3. The method of claim 2 further comprising:
20 searching an attribute mapping data structure using
the object identifier in the received parameter;
in response to a determination of a matching object
identifier in the attribute mapping data structure,
retrieving a class identifier associatively stored with
25 the matching object identifier in the attribute mapping
data structure; and
calling a second object-oriented method in a class
identified by the retrieved class identifier.

30 4. A method in a data processing system for managing
data attributes, the method comprising the steps of:

Docket No. AUS990800US1

invoking a first object-oriented method to process an attribute object, wherein the first object-oriented method is defined in an abstract class for attribute objects with a subclass for undefined attributes and a

5 subclass for defined attributes, wherein the subclass for defined attributes is further comprised of a subclass for each PKCS-defined (Public Key Cryptography Standards) attribute and a subclass for each user-defined attribute;

invoking a second object-oriented method to process

10 an attribute object, wherein the second object-oriented method is defined in a PKCS9 gateway class; and

in response to invoking the first object-oriented method or the second object-oriented method, processing the result returned by the first object-oriented method

15 or the second object-oriented method.

5. The method of claim 4, where a PKCS compatible attribute is a unique object identifier and value, as defined by the Abstract Syntax Notation (ASN.1) for the

20 X.500 Attribute type.

6. The method of claim 4, wherein each defined attribute is implemented as a separate class.

25 7. The method of claim 4, wherein each defined attribute is registered with the PKCS9 gateway class.

8. The method of claim 7, wherein PKCS-defined attributes are registered statically with the PKCS9 gateway class.

30

9. The method of claim 7, wherein the user-defined attributes are registered with the PKCS9 gateway class by

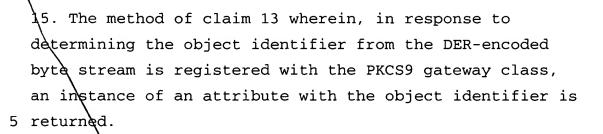reading a configuration file when the PKCS9 gateway class is initially loaded.

10. The method of claim 4, wherein the second
5 object-oriented method determines a type of the attribute object by performing an instanceof comparison to registered attributes.

11. The method of claim 4, wherein the attribute object
10 is constructed using a constructor method in a class associated with a PKCS-compatible attribute.

12. The method of claim 4 wherein, in response to determining an object identifier and a value associated
15 with the object identifier and determining the object identifier is registered with the PKCS9 gateway class, the PKCS9 gateway class returns an instance of a registered attribute.

20 13. The method of claim 4 wherein, in response to determining a DER-encoded byte stream with an object identifier, the second object-oriented method in the PKCS9 gateway class returns an instance of a PKCS-compatible attribute.
25

14. The method of claim 13 wherein, in response to determining the object identifier from the DER-encoded byte stream is not registered with the PKCS9 gateway class, an instance of an undefined attribute is returned
30 with the value being a DER-encoded byte stream.

Docket No. AUS990800US1

15. The method of claim 13 wherein, in response to determining the object identifier from the DER-encoded byte stream is registered with the PKCS9 gateway class, an instance of an attribute with the object identifier is
5 returned.

16. The method of claim 4, wherein a registered attribute object is encoded to a DER-encoded byte stream by using the first object-oriented method for encoding the
10 attribute object.

17. The method of claim 4, wherein a registered attribute object represented as a DER-encoded byte stream is decoded to an attribute object by using the second
15 object-oriented method for decoding the attribute object.

18. The method of claim 4, wherein a second object-oriented method in the PKCS9 gateway class extracts attribute values into forms, wherein the forms
20 are strings, numbers, and/or other non-abstract data types

19. A data processing system for managing data attributes, the data processing system comprising:
25   registering means for registering attributes with a PKCS9 gateway class, wherein the attributes include user-defined attributes and PKCS-standard (Public Key Cryptography Standards) defined attributes;
    storing means for associatively storing an
30 identifier for each of said attributes.

Docket No. AUS990800US1

20. The data processing system of claim 19 further comprising:

calling means for calling a first object-oriented
5 method in the PKCS9 gateway class, wherein the first
object-oriented method receives a parameter comprising an
object identifier for an attribute.

21. The data processing system of claim 20 further
10 comprising:

searching means for searching an attribute mapping
data structure using the object identifier in the
received parameter;

retrieving means for retrieving, in response to a
15 determination of a matching object identifier in the
attribute mapping data structure, a class identifier
associatively stored with the matching object identifier
in the attribute mapping data structure; and

calling means for calling a second object-oriented
20 method in a class identified by the retrieved class
identifier.

22. A data processing system for managing Public Key
Cryptography Standards (PKCS) compatible attributes, the
25 data processing system comprising:

first constructing means for constructing a new
instance of an attribute object;

first differentiating means for differentiating
between attribute objects of different types;

30   converting means for converting an instance of an
attribute object to and/or from DER-encoding;

Docket No. AUS990800US1

first extracting means for extracting values
associated with an attribute object;

extending means for extending a set of attributes
with user-defined types; and

5   first registering means for registering an attribute
class with a PKCS9 gateway class.

23.   The data processing system of claim 22, where a PKCS
compatible attribute is a unique object identifier and
10 value, as defined by the Abstract Syntax Notation (ASN.1)
for the X.500 Attribute type.

24.   The data processing system of claim 22 further
comprising:
15   an abstract attribute object class with an undefined
attribute object subclass and a defined attribute object
subclass, wherein the defined attribute object subclass
is further comprised of a subclass for each PKCS-defined
attribute object and a subclass for each user-defined
20 attribute object.

25.   The data processing system of claim 22 further
comprising:
second constructing means for constructing wherein a
25 new instance of an attribute object using a class
constructor.

26.   The data processing system of claim 22 further
comprising:
30   third constructing means for constructing a new
instance of a PKCS-compatible attribute object is
constructed using the PKCS9 gateway class if an attribute

Docket No. AUS990800US1

object identifier and a class implementing that attribute
are registered.

27. The data processing system of claim 22 further
5 comprising:
fourth constructing means for constructing a new
instance of a PKCS-compatible attribute object using the
PKCS9 gateway class based on a DER-encoded byte stream.

10 28. The data processing system of claim 22 further
comprising:
second differentiating means for differentiating a
type for an attribute object determined by performing an
instanceof comparison to registered attribute classes.
15

29. The data processing system of claim 22 further
comprising:
encoding means for encoding an attribute object to a
DER-encoded object by using an encode method of the
20 attribute object.

30. The data processing system of claim 22 further
comprising:
decoding means for decoding an attribute object
25 represented as a DER-encoded string to an attribute
object by using a decode method of the attribute object.

31. The data processing system of claim 22 further
comprising:
30 second extracting means for extracting the PKCS9
gateway class returns attribute values, wherein the

Docket No. AUS990800US1

values are represented as strings, numbers, and/or other
non-abstract data types.

32.   The data processing system of claim 22 further
5 comprising:
          second registering means for registering a
PKCS-compatible attribute is registered with the PKCS9
gateway class.

10 33.   The data processing system of claim 32 further
comprising:
          third registering means for registering an attribute
defined by the Public Key Cryptography Standards is
registered with the PKCS9 gateway class.
15
34.   The data processing system of claim 32 further
comprising:
          fourth registering means for registering wherein
user-defined attributes are registered with the PKCS9
20 gateway class by reading a configuration file when the
PKCS9 gateway class is initially loaded.

      35.   A computer program product in a computer-readable
medium for use in a data processing system for managing
25 data attributes, the computer program product comprising:
          first instructions for registering attributes with a
PKCS9 gateway class, wherein the attributes include
user-defined attributes and PKCS-standard (Public Key
Cryptography Standards) defined attributes;
30   second instructions for associatively storing an
identifier for each of said attributes.

Docket No. AUS990800US1

36. The computer program product of claim 35 further comprising:

   instructions for calling a first object-oriented
5 method in the PKCS9 gateway class, wherein the
object-oriented method receives a parameter comprising an
object identifier for an attribute.

37. The computer program product of claim 36 further
10 comprising:

   instructions for searching an attribute mapping data
structure using the object identifier in the received
parameter;

   instructions for retrieving, in response to a
15 determination of a matching object identifier in the
attribute mapping data structure, a class identifier
associatively stored with the matching object identifier
in the attribute mapping data structure; and

   instructions for calling a second object-oriented
20 method in a class identified by the retrieved class
identifier.

38. A computer program product on a computer-readable
medium for use in a data processing system for managing
25 Public Key Cryptography Standards (PKCS) compatible
attributes, the computer program product comprising:

   instructions for constructing a new instance of an
attribute object;

   instructions for differentiating between attribute
30 objects of different types;

   instructions for converting an instance of an
attribute object to and from DER-encoding;

Docket No. AUS990800US1

instructions for extracting values associated with
an attribute object; and

instructions for extending a set of attributes with
user-defined types; and

5   instructions for registering an attribute class with
a PKCS9 gateway class.

39.   The computer program product of claim 38, where a
PKCS compatible attribute is a unique object identifier
10 and value, as defined by the Abstract Syntax Notation
(ASN.1) for the X.500 Attribute type.

40.   The computer program product of claim 38, wherein
an abstract attribute object class with an undefined
15 attribute object subclass and a defined attribute object
subclass, wherein the defined attribute object subclass
is further comprised of a subclass for each PKCS-defined
attribute object and a subclass for each user-defined
attribute object.
20

41.   The computer program product of claim 38, wherein a
new instance of a PKCS-compatible attribute object is
constructed using a class constructor.

25 42.   The computer program product of claim 38, wherein a
new instance of a PKCS-compatible attribute object is
constructed using the PKCS9 gateway class if an attribute
object identifier and a class implementing that attribute
is registered.
30

43.   The computer program product of claim 38, wherein a
new instance of a PKCS-compatible attribute object is

constructed using the PKCS9 gateway class based on a
DER-encoded byte stream.

44.   The computer program product of claim 38, wherein a
5 type for an attribute object is determined by performing
an instanceof comparison to registered attribute classes.

45.   The computer program product of claim 38, wherein an
attribute object is encoded to a DER-encoded object by
10 using an encode method of the attribute object.

46.   The computer program product of claim 38, wherein an
attribute object represented as a DER-encoded string is
decoded to an attribute object by using a decode method
15 of the attribute object.

47.   The computer program product of claim 38, wherein
the PKCS9 gateway class returns attribute values, wherein
the values are represented as strings, numbers, and/or
20 other non-abstract data types.

48.   The computer program product of claim 38, wherein a
PKCS-compatible attribute is registered with the PKCS9
gateway class.
25

49.   The computer program product of claim 48, wherein an
attribute defined by the Public Key Cryptography
Standards is registered with the PKCS9 gateway class.

30 50.   The computer program product of claim 48, wherein
user-defined attributes are registered with the PKCS9

Docket No. AUS990800US1

gateway class by reading a configuration file when the PKCS9 gateway class is initially loaded.